Computer vision for image-based transcriptomics

Stoeger, Thomas ; Battich, Nico ; Herrmann, Markus D ; Yakimovich, Yauhen ; Pelkmans, Lucas

Abstract: Single-cell transcriptomics has recently emerged as one of the most promising tools for understanding the diversity of the transcriptome among single cells. Image-based transcriptomics is unique compared to other methods as it does not require conversion of RNA to cDNA prior to signal amplification and transcript quantification. Thus, its efficiency in transcript detection is unmatched by other methods. In addition, image-based transcriptomics allows the study of the spatial organization of the transcriptome in single cells at single-molecule, and, when combined with superresolution microscopy, nanometer resolution. However, in order to unlock the full power of image-based transcriptomics, robust computer vision of single molecules and cells is required. Here, we shortly discuss the setup of the experimental pipeline for image-based transcriptomics, and then describe in detail the algorithms that we developed to extract, at high-throughput, robust multivariate feature sets of transcript molecule abundance, localization and patterning in tens of thousands of single cells across the transcriptome. These computer vision algorithms and pipelines can be downloaded from: https://github.com/pelkmanslab/ImageBasedTranscriptomics.

DOI: https://doi.org/10.1016/j.ymeth.2015.05.016

Computer Vision for Image-Based Transcriptomics

Thomas Stoeger[a,b,1], Nico Battich[a,b,1], Markus D. Herrmann[a,b], Yauhen Yakimovich[a], Lucas Pelkmans[a,2]

[a] Faculty of Sciences, Institute of Molecular Life Sciences, University of Zurich, Winterthurerstrasse 190, CH-8057 Zurich, Switzerland.

[b] Life Science Zurich Graduate School, Ph.D. program in Systems Biology.

[1] These authors contributed equally to this work.

[2] Corresponding author: L.P. lucas.pelkmans@imls.uzh.ch

Abstract

Single-cell transcriptomics has recently emerged as one of the most promising tools for understanding the diversity of the transcriptome among single cells. Image-based transcriptomics is unique compared to other methods as it does not require conversion of RNA to cDNA prior to signal amplification and transcript quantification. Thus, its efficiency in transcript detection is unmatched by other methods. In addition, image-based transcriptomics allows the study of the spatial organization of the transcriptome in single cells at single-molecule, and, when combined with superresolution microscopy, nanometer resolution. However, in order to unlock the full power of image-based transcriptomics, robust computer vision of single molecules and cells is required. Here, we shortly discuss the setup of the experimental pipeline for image-based transcriptomics, and then describe in detail the algorithms that we developed to extract, at high-throughput, robust multivariate feature sets of transcript molecule abundance, localization and patterning in tens of thousands of single cells across the transcriptome. These computer vision algorithms and pipelines can be downloaded from: https://github.com/pelkmanslab/ImageBasedTranscriptomics

1. Image-based transcriptomics is unique in several ways

In the past few years a wealth of techniques have been developed to study genome-wide transcriptional output at the single-cell level [1-7]. In contrast to methods relying on sequencing or PCR, image-based transcriptomics visualizes single transcripts in a population of single cells *in situ*. This allows not only the absolute quantification of transcript copy numbers, but also the

spatial mapping of transcript molecules to the sub-cellular microenvironment [4]. Being an *in situ* technology, it does not require homogenization of cells and therefore minimizes the loss of material, thus achieving very high detection efficiency [4]. Another advantage of image-based transcriptomics is that it can be combined with the phenotypic characterisation of each single cell and its context within a population of cells or tissue, by microscopic assays and stainings commonly used in cell and developmental biology. This makes image-based transcriptomics of particular interest when studying the localization dynamics of the transcriptome in response to stimuli or perturbations and to identify sources of cell-to-cell variability in these processes [8,9]. While establishing image-based transcriptomics, we soon realized that a robust computer vision pipeline was as important as the experimental platform for accurately identifying and characterizing each single transcript molecule within a cell. Therefore, we here describe in detail our recent computer vision algorithms that result in accurate detection of objects in spinning disk confocal microscopy images. Besides providing a robust guide for identifying billions of individual transcript molecules with little hands-on user time, we describe how to unlock functionally important parameters of gene expression, which are impossible to grasp without the power of computer vision. For instance, multivariate descriptors of the position of each single transcript molecule enable an unsupervised characterization of the localization of transcripts of every cell.

1.1. General outline

Image-based transcriptomics employs multi-well plates to stain cells in parallel with specific probes against a transcript of interest (Fig. 1). Within single wells of a multi-well plate, the transcripts of different genes are stained by an automated experimental procedure. Each single transcript molecule is detected by high-throughput microscopy and computer vision. Experimental and computational steps can be performed with equipment that is commonly used for image-based high-throughput assays.

Each single transcript molecule is stained by branched DNA single-molecule *in situ* hybridization (bDNA sm-FISH). This technology, which is commercially available from Affymetrix and Advanced Cell Diagnostics, applies a series of consecutive *in situ* hybridizations, which visualize each single transcript molecule as a bright fluorescent spot. In a first round of *in situ* hybridization, two epitope-specific primary probes bind next to each other on the same transcript molecule. While it is technically possible to implement bDNA FISH with only one epitope-specific probe [10], requiring the simultaneous binding of two probes in direct spatial adjacency

should reduce unspecific signal [11]. Targeting 15 different epitopes of each transcript in a single hybridization reaction ensures that at least one epitope is accessible to the detection reagents without the need to denature the specimen. The subsequent rounds of *in situ* hybridization create a docking platform for ~500 fluorescently labelled probes per single epitope. This level of fluorescence is sufficiently high to enable the specific, rapid and robust detection of single transcript molecules by high-throughput imaging.

1.2. Alternative methods for RNA detection in imaging

Another method for directly visualizing single transcript molecules *in situ* is oligonucleotide-based single molecule FISH (o-nuc sm-FISH). This approach targets individual transcripts by up to 40 different oligonucleotides, which are directly conjugated to fluorophores. While a recent study achieved to monitor 61 different ncRNAs, it had to restrict itself to "a few dozen cells … due to limited imaging throughput" [12]. Possibly, this reflects the lower signal-to-noise ratio of single fluorescent spots of o-nuc sm-FISH and their need for a 600 times longer illumination time [4].

Alternatively, transcripts can be visualized indirectly via reverse transcription to cDNA that can be sequenced *in situ* by padlock probes [13] or oligonucleotide ligation and detection [14,15]. While the former sequencing approach can presently detect 31 different genes simultaneously in thousands of single cells within a tissue slide [13], the latter approach can currently read around 200 mRNAs simultaneously for 40 different cells [15]. The efficiency for detecting single transcript molecules has been estimated to be 30% [13,16] and 3% [15] respectively, which is much lower than the 85% of hybridization efficiency in sm-FISH [4,17]. Such low efficiencies currently prevent these alternative methods from surveying the transcriptome with single-molecule sensitivity and resolution *in situ* [18,19].

2. Establishing image-based transcriptomics with single molecule resolution

The detailed experimental protocol for high-throughput bDNA sm-FISH has been published previously [4] and therefore, we here mainly provide additional assistance for setting up a robust automated experimental platform. As a general introduction to high-throughput image-based assays and the infrastructure and software supporting such experiments we highly recommend the excellent essay by Buchser and colleagues [20].

Table 1 contains an overview of potential problems occurring during the detection of single transcripts. The most critical factor in getting reliable results is to use an automated incubator that contains rotating towers for the individual storing of multi-well plates during hybridization reactions. This prevents the occurrence of different hybridization efficiencies in different wells of a multi-well plate (data not shown). Table 2 highlights potential pitfalls, which could affect the biological interpretation of accurate single-molecule measurements. We recommend repeating the control experiments suggested in Table 1 and Table 2 in different weeks to ensure that your setup of image-based transcriptomics functions robustly.

Possible artefact	**Experiment**	**Hints**
Inability to detect single molecules	Assay with probes against a single epitope of HPRT1 [4].	exposure time during imaging; protease concentration
False positive detection	Probe against bacterial gene dapB. Less than 1 spot per cell should be detected.	protease concentration; cells without cytoplasmic DNA
Spill-overs	Stain adjacent wells for the negative control (bacterial dapB) and the highly abundant ACTB transcripts. Test full plates.	liquid handling
Efficiency of single molecule detection	Stain same transcript on two different sets of epitopes by two different sets of amplification reagents, which can be visualized by two different fluorophores. Efficiency of detection should be ~85% [4].	protease concentration; amount of targeted epitopes per transcript; computational spot detection
Positional bias between wells (staining reaction)	Stain all wells of a multi-well plate with probes against the non-abundant housekeeping gene HPRT1.	always use incubator with rotating towers for hybridization; never skip protocolled in-solution mixing
Low reproducibility	Multiple independent assays across different weeks.	aberration of liquid handling <1%; cell seeding
Tearing of signal of single molecules	Compare signal obtained by multiple units and types of objectives.	choose best objective and remove remaining effect computationally (see below)

Table 1: Suggested controls for the detection of transcript molecules.

Possible artefact	**Experiment**	**Hints**
Positional bias between wells (biological)	Compare the number of cells and local cell density [21] across individual wells.	avoid “edge-effects” by following the cell seeding protocol of Lundholt et al. [22]
Variable number of cells per seed	Monitor and potentially adjust cell dissociation protocol such that, on average, a cell aggregation score of less than 1.2 is achieved for each seed.	trypsinization time; repeatedly shear cells through pipette pressed towards plastic dish
Loss of cells during assay	Perform live-imaging of cells with Hoechst dye prior to the assay and compare with presence of cells after image-based transcriptomics.	slowly pipet to side of well (most steps) or center of well (only for in-solution mixing)

Staining of cell-outline varies between experiments	Repeat and time succinimidyl ester staining with multiple freshly prepared staining solutions.	time-dependent decay of carboxylic acid, succinimidyl ester, in aqueous solutions

Table 2: Suggested controls for the proper interpretation of single molecule measurements.

3. Establishing the image analysis pipeline

A robust image analysis pipeline is required for accurate measurements of absolute transcript levels as well as measurements of transcript localization in the cytoplasm of single cells, and extraction of features that describe the cellular phenotype. First, homogeneous intensity values throughout the images in all channels must be ensured, and then object segmentation must be performed minimizing errors. To ensure this, we developed four algorithms to perform high-throughput illumination correction of raw images, robust nuclei and cell segmentation, and robust spot detection. They can be downloaded from https://github.com/pelkmanslab/ImageBasedTranscriptomics and applied on an example dataset available on https://image-based-transcriptomics.org

The algorithms presented in this manuscript do not intend to replace single-cell quality control. For the latter we recommend interactive user-guided supervised machine learning, which has been implemented before by our group [23] and others [24]. Supervised machine learning not only readily identifies rare cells that have not been correctly segmented, but also allows the selection of a group of cellular objects that is relevant for a specific biological question (e.g.: interphase cells).

The algorithms presented in this manuscript intend to reduce human hands on time and increase the amount of high-quality primary data after computational image-analysis (Table 3). Computational running time has not emerged as a practical issue for image-based transcriptomics. The algorithms are robust in the sense that their input parameters rarely have to be adjusted for individual experimental plates.

	Hands-on time	Computational time	Computers used by us
Illumination correction (this manuscript)	5min	2-5h	4
Nucleus segmentation (CellProfiler)	30min	<< 1h	1500

Nucleus segmentation (this manuscript)	30min	<< 1h	1500
Cell segmentation (CellProfiler)	5-10h	<< 1h	1500
Cell segmentation (this manuscript)	5min	2-10h	1500
Cell segmentation (manual segmentation)	>> 1 month (expected)		
Inference of spot detection parameters (this manuscript)	1h-2h	2-8h	10
Optional lens aberration correction	1h	1h-2h	1500
Spot detection (this manuscript)	30min	<<1h	1500
Measuring localization of transcripts	5min	1h-2h	1500

Table 3: Time for image-based transcriptomics on ten 384-well plates to obtain results, whose quality appeared acceptable to us. Time estimates are based on our experience and depend upon the specific computational infrastructure.

While the principles of the algorithms presented in this manuscript have been sketched in one of our earlier publications, the description beneath provide a detailed guide for using those algorithms. Moreover we here include implementations of these algorithms for MATLAB and implementations as modules for CellProfiler to segment single nuclei and cells.

3.1. Illumination correction

Illumination correction of raw images is essential for subsequent steps in the image analysis pipeline. It ensures correct object detection and accurate measurements of intensity features, reducing biases due to uneven illumination of the sample as well as positional differences in the signal gain resulting from the detection system. During image-based transcriptomics, we exploit the statistical power of the large number of images acquired per channel to learn pixel-wise illumination and signal gain biases (Battich et al. 2013 [4], Fig 2). Briefly, we calculate the standard deviation and mean intensity values per pixel for every pixel position of a given channel. To correct the illumination bias, per-pixel z-scoring is performed as shown in Fig. 2(Eq 1). The z-scored values are then reversed to intensity values as shown in Fig. 2(Eq 2).

3.2. Nucleus segmentation

Pixels belonging to nuclei objects can be easily distinguished from background pixels by thresholding an image of a nuclei-specific stain such as DAPI. However, this often results in clumps of several nuclei because a single, image-wide threshold value is generally not sufficient to separate nuclei that lie very close to each other. Such clumped objects are relatively large and display multiple concave regions. Generally, at the intersection of individual objects, a line of low intensity pixels connects two concave regions, which can be found by the watershed algorithm [25]. Thus, we identify single nuclei with an algorithm consisting of two parts: first, intensity thresholding by the Otsu method [26] identifies primary objects; and, secondly, objects consisting of multiple nuclei are separated along the best identified watershed line (Fig. 3).

The algorithm (algorithm 1) uses illumination-corrected images and processes them as follows:
1) Initial objects are identified by simple thresholding.
2) Clumped objects are selected on the basis of size and shape features: area, solidity, and form factor.
3) The perimeter of selected objects is analysed and concave regions along the boundary of objects are identified.
4) Putative watershed lines connecting two concave regions are determined using the Dijkstra shortest-path algorithm [27].
5) All possible cuts along the selected watershed lines are considered and features of each potential cutting line (intensity along the line, angle between concave regions) as well as features of the resulting objects (area/shape) are measured.
6) An "optimal" cut line is finally chosen by minimizing a cost function that evaluates the measured features. The resulting objects have a minimal size and are as round as possible, while the separating line is as straight and as short as possible and the intensity along the line as low as possible.

Algorithm 1 *IdentifyPrimaryIterative*

```
1. Initialize() // initialize objects by thresholding the input intensity image;
2. InitialSegmentation() // recognize objects as segmented objects without cutting them first;
3. Repeat
4.         For each object in segmented objects
5.                 If  lower size threshold < size of object < upper size threshold
6.                         and solidity of object < solidity threshold
7.                         and transformed form factor of object > form factor threshold
8.                 then
9.                    Add object to the collection of clumped objects to be cut;
```

```
10.             end
11.         End
12.         PerimeterAnalysis()  // analyze perimeter of selected clumped objects and calculate
                                 the curvature along their boundary [see PerimeterAnalysis.m];
13.         PerimeterWatershedSegmentation() // cut selected clumped objects along watershed lines
                                             between concave regions
                                             [see PerimeterWatershedSegmentation.m];
14.         For each object in clumped objects
15.             IdentifyConcaveRegions() of the object, where region is concave
16.               If angle of circle segment of region > equivalent segment threshold
17.               and radius of region < equivalent radius threshold
18.             IdentifyLinesAndNodes() of the object // find all watershed lines and nodes, where each
                                                      node is a single pixel on the line that overlaps
                                                      with the object boundary;
19.             Select watershed nodes If node lies within concave regions;
20.             Select all watershed lines If line connects two watershed nodes
                                            from different concave regions.;
21.             For each line in watershed lines
22.                 Measure line length and straightness and the intensity profile along the line;
23.                 Measure the angle between normal vectors at watershed nodes;
24.                 Measure area, solidity and form factor of the cut object, i.e. the smaller of the two
                        objects that would result from a cut along the line.
25.                 If size of the object < threshold of object being too small
26.                 then
26.                     discard such cutting line from selected watershed lines;
27.                 end
28.                 Select "optimal" watershed line by minimizing the cost function
29.                     cost(a, b, c, d, e, f, g, h) ← a - 2*b - c - d - e + 2*f - g - 2*h,
30.                 where
                        a is a solidity of cut object,
                        b is a form factor of cut object,
                        c is a mean intensity along the line,
                        d is a max intensity along the line,
                        e is a 0.75 quantile intensity along the line,
                        f  is an angle between two watershed nodes,
                        g is a line straightness,
                        h is a line length.
31.             End // of for-each-loop at line 21
32.         End // of for-each-loop at line 14
33. Until no more clumped objects can be found.
```

Whenever attempting to identify individual nuclei of a novel cell line or whenever changing imaging conditions, we recommend to empirically test different schemes and parameters for segmentation of nuclei. Good settings can usually be found empirically by using the inbuilt testing mode of IdentifyPrimaryIterative. Contrasting CellProfiler's default options for separating objects, which are part of the IdentifyPrimaryAutomatic module, IdentifyPrimaryIterative can simultaneously consider the local intensity of the DAPI stain and the geometry of identified objects to separate them. In practice we never had to adjust the threshold value suggested by

the Otsu method [26]. Depending on the biological question of interest, one might choose settings for the separation of objects, which favour over- or under-segmentation. For instance over-segmentation increases the fraction of emerging cells during anaphase cells that are already considered as individual objects. Under-segmentation on the other hand facilitates the correct segmentation and thus quantification of multinucleate cells.

Frequently not every object, which can be identified in image-based assays, should be considered in subsequent analysis. For instance we preclude the analysis of DAPI positive cellular debris, apoptotic bodies and mitotic cells by a dual strategy, which is independent of IdentifyPrimaryIterative. First the DiscardObjectsBySize.m module removes small objects within CellProfiler. Second, supervised machine learning identifies debris, and apoptotic and mitotic cells [23].

3.3. Cell segmentation

The segmentation of cells uses the segmentation of nuclei as seeds [28]. It is imperative to ensure correct segmentation of the cellular cytoplasm as this will not only have a major impact in the number of spots (or transcript molecules) allocated to each cell, but will also drastically affect measurements of transcript localization. To achieve the high accuracy in cell segmentation required for image-based transcriptomics, we developed an algorithm that performs sequential rounds of watershedding, rather than the one round of watershedding typically applied [28]. This iterative algorithm allows accurate identification of the boundary between cells with relatively minimal user input.

In the algorithm, an arbitrary amount of different segmentations are combined in such a way that the allocation of single pixels to their correct seeds (nuclei) never becomes worse and thus becomes optimal by iteratively performing many different segmentations (Fig. 4). Besides largely eliminating human hands-on time, this strategy generally yields superior results compared to a single segmentation: different parts of a single cell can be segmented by opposing segmentation settings, which only yield optimal segmentation accuracy in a subpart of the cell, but perform sub-optimally in other subparts.

Briefly, the algorithm (Algorithm 2) treats the input images as follows:
1) Calculate the watershed cell segmentation at different thresholds.

2) One label image is constructed. If a pixel is part of different objects at a given threshold (which is likely in cell-rich regions), it will be allocated to the object of the higher threshold (e.g. if threshold specifications were 1 and 0.5, it would be attributed to the object identified with a threshold of 1).

3) Define background pixels by a single user-provided microscope-specific threshold, which can be determined manually once.

4) Re-label pixels of prospective objects (cells), which are not connected to their original seed (nucleus), as pixels belonging to the background.

Algorithm 2 *IdentifySecondaryIterative*

1. Initialize empty *FinalSegmentation* matrix;
2. Load *OrigInputImage* matrix;
2. *DefineThresholds(OrigInputImage)* // defines a sorted list of *thresholds* {T_i}, where T_{min} is a minimal threshold [e.g. choosen by CPthreshold.m] and $T_i < T_{i+1}$.
3. *SeedMarkersImage*← *DefineSeedObjects(OrigInputImage)* // labels each pixel with grayscale values, uniquely enumerating each *seed object* (e.g. nuclei) by its ID (usually, a foreground mask from previous segmentation);
4. **For each** *threshold* T_i **in** *thresholds* {T_i}
5. Obtain binary *ThresholdedImage* of pixels, **where** each *pixel* = 1 **If** pixel intensity > T_i **and** *pixel* **not in** *foreground* **else** *pixel* = 0;
6. Find labeled segmentation S_i ←*WatershedMethod*(*ThresholdedImage, SobelGradient(OrigInputImage)*, SeedMarkersImage); // [see *IdentifySecondary.m for 'Watershed'* choice of method];
7. **Select** *indexes* of all non-background pixels **in** segmentation S_i;
8. *FinalSegmentation(indexes)* ← S_i *(indexes); //* overwrite selected pixels within FinalSegmentation with the label values in S_i;
9. **End**
10. *CleanSegmentation(FinalSegmentation)* // Make sure *FinalSegmentation* complies to CellProfiler expectations.

function *CleanSegmentation(FinalSegmentation):*

1. Identify borders between different object labels as the non-zero values upon Sobel-filtering, i.e. *SobelGradient*(*FinalSegmentation)*;
2. Set the identified borders between different label values of objects to background;
3. Set pixels with the object labels to background value, if other pixels with the same object labels do not connect them to the seed with the same label values of objects.

In our experience IdentifySecondaryIterative has never been performing worse than segmentation by a single round of wathershedding. The few remaining miss-segmented cells

can identified by supervised machine learning. As with any image-based assay the ability resolve fine structures of the cellular periphery depends upon their size and the resolution of the microscopic images. Like other algorithms that segment 2D images to segment cells, IdentifySecondaryIterative works best on cells that do not grow on top of each other, such as HeLa Kyoto cells, REPI cells and primary Keratinocytes. If cells can grow on top of each other, it is not always possible to allocate a single pixel to a single cell (e.g.: A431, NIH 3T3, HEK), though supervised machine learning could be used to discard those cells, which grow on top of each other.

3.4. Spot detection and correction of lens aberrations

The basic strategy for detecting single transcripts as spots has been developed by Jiri Matas [32] and Arjun Raj and their colleagues [17]. After emphasizing spot-like signal by a Laplacian of Gaussian filter (Fig. 5A), a threshold for the detection of objects is chosen such that, on each single image, the specific value of the threshold only mildly affects the number of detected transcripts (Fig. 5B,C). As the numerical value of the threshold will partially depend upon the absolute intensity of the acquired images, we rescale the intensities of individual images such that they are comparable between different images and a single numerical value for the threshold can be chosen (Fig. 5C). This seemingly minor, but essential, detail of our image-analysis pipeline contrasts the most common high-throughput implementation of spot detection algorithms, which rescales the intensities of any image according to the intensities of its dimmest and brightest pixel [17,28,33]. While the accompanying code supports additional refinement of the spot detection, these additional parameters (2D/3D, minimal intensity of pixels, size of spots) have a negligible effect on the detection of transcripts once robust imaging conditions have been established experimentally.

For identifying the settings for detecting spots, we include on each experimental plate 4 wells in which bacterial dapB transcripts are probed (a negative control for mammalian cells), and 4 wells for probing transcripts of the housekeeping gene HPRT1 (Hypoxanthine Phosphoribosyltransferase 1, which plays a central role in purine nucleotide synthesis). In a first computational step, we find the upper- and lower-image intensity boundaries for those two reference transcripts (see Exp_getIntensitiesOfReferences.m). The next step detects spots at varying threshold values while rescaling the intensities of single images according to the previously identified bounds (see Exp_getSpotCountsOfReferences.m). Upon completion of the computation, a threshold is chosen manually such that its specific numerical value only mildly

affects the number of detected transcripts (see Exp_selectDetectionThreshold.m). In practice, a fast manual choice and optimization of settings is as good as a fully computational procedure, but offers the advantage of being a first quality control of the data. The number of spots in the dapB negative control should be much lower and more sensitive towards changes in the numerical value of the threshold [4].

Optical aberrations, which tear the signal of individual transcript molecules in the corners of an image, make the signal less spot-shaped. This creates a spatial bias in the detection of transcript molecules of approximately +/- 3% at different positions of an image [4]. While it is best to reduce this effect experimentally (see above), it can be optionally attenuated further by computationally modifying the threshold for the spot detection at different positions of an image. Use the ScanSpotThresholds.m CellProfiler module to test multiple different thresholds surrounding the previously identified reference threshold. Inclusion of all images of a plate (recommended: approximately 10,000 images), allows computing the spatial bias of the detection of spots, which can be used to construct a correction matrix that will modify the spot detection threshold for each pixel (see Exp_computeCorrectionMatrix.m).

You can now identify spots with a CellProfiler pipeline containing the IdentifySpots2D.m module; optionally apply a correction matrix against the spatial bias, which can be loaded by the LoadSingleMatrix.m module; and, insert the parameters for the spot detection determined above. Additionally, we recommend enabling the deblending option, an algorithm from astrophysics [34], which can spatially resolve individual transcript molecules below the optical diffraction limit. If a correction matrix for the spatial bias has been applied, monitor its impact on the spatial bias of the spot detection (see Exp_checkBiasCorrection.m) and potentially restrict or expand the range of thresholds that have been considered for the construction of the correction matrix.

In addition to the algorithm outlined above, which provides highly reproducible and specific measurements of the number of transcripts in a high-throughput experimental setup with bDNA sm-FISH[4], we would like to note several excellent algorithms, that have been used with o-nuc sm-FISH to identify those fluorescent spots that likely indicate single transcripts [29-31].

3.5. Quantification of spot localization

Being an *in situ* technology, image-based transcriptomics can quantify the localization of each single transcript molecule. Although the subcellular localization of transcripts and it variability

across single cells can hold more biological information than single-cell transcript abundance [4], it is not yet used routinely in functional genomics studies due to technical limitations. This section describes how this powerful source of information can be unlocked from image-based transcriptomics data.

Each single transcript molecule can be characterized by a set of measurements (Algorithm 3), which describe its distance to the centroid or edge of an organelle or the cell [4]. In addition, the position of each transcript molecule can be placed in relation to other molecules, for instance by measuring the variance of its pairwise distances to all other molecules, or by counting the number of transcript molecules within a certain area. Such readouts of single molecules are created by the MeasureLocalizationOfSpots.m CellProfiler module [4]. By choosing an arbitrary amount of differently sized areas, different scales of subcellular crowding can be compared.

Algorithm 3 *CPgetSpotLocalizations(LookupImage, VectorWithDistancesForFractions, VectorWithDistanceContainingFractionOfSpots)*

```
1.  Initialize Results  // a key-value array, containing all measurements;
2.  Define SpotDistances as all Euclidean distances between spot pairs (cartesian product);
3.  // Determine fractions of spots within given distance and distances for given fractions of spots;
4.  For each spot in spots
5.      For each DistanceOfFraction in VectorWithDistancesForFractions
6.          Results[FractionOfSpotsAtDistance] ← normalize over
7.              Select all spots within given DistanceOfFraction exclude the spot itself;
8.      End
9.      For each DistanceContainingFractionOfSpots in
VectorWithDistanceContainingFractionOfSpots
10.         Results[DistanceContainingFractionOfSpots] ← Select min(
11.             all SpotDistances for a given spot within DistanceContainingFractionOfSpots
                                                    exclude the spot itself);
12.     End
13. End
14. Results[MeanDistance] ← mean(columns of SpotDistances );
15. Results[VarianceDistance] ← variance(columns of SpotDistances );
16. Results[StandardDeviationDistance] ← sqrt(ResultsVarianceDistance);
17. Results[DistanceToCellCentroid] ← measure distances of all spots to centroid of the cell;
18. // Treat spots at the cell membrane specially.
19. For each spot in spots
20.     Determine coordinate of the closest membrane pixel;
21.     Results[ShortestDistanceToMembrane] ← EuclidianDistance(centroid of spot,
                                                    closest membrane pixel);
22.     Results[DistanceToNucleus] ← EuclidianDistance(centroid of spot, centroid of nucleus);
23.     If EuclidianDistance(centroid of spot, closest membrane pixel) > sqrt(2) then
            // spot is not at the membrane;
            Construct a projection line connecting the centroid of the nucleus and centroid of the spot;
24.         Results[DistanceAlongProjection] ← EuclidianDistance(centroid of spot,
```

```
                    closest membrane pixel) // membrane pixel is picked along the projection line;
25.     else
26.         Results[DistanceAlongProjection] ← Results[ShortestDistanceToMembrane]
27.     end
28. End
29. Results[MembraneBorderingCell] ← look up pixel at position within LookupImage
                                     // LookupImage is an image indicating for each pixel,
                                        whether closest membrane is adjacent to a cell);
30. return Results.
```

Cellular readouts of transcript localization can be derived from the readouts of single transcript molecules. For instance, one may compute the first central moments of the distribution of every readout across all single transcript molecules within a single cell with the accompanying MeasureChildren.m CellProfiler module [4], and subsequently quantify properties of the single-cell distributions of these central moments. In practice, these information-rich multivariate readouts for each single cell, generated for thousands of cells in a single population, rarely lend themselves to ready interpretation or presentation. Therefore, we have previously developed and documented [4] an unsupervised clustering scheme that uses selected cellular statistics to identify a small number of main patterns in single cell subcellular transcript localization. This analysis has been well described by us [4] and can be computed independently of CellProfiler by our locpatterns package (https://github.com/pelkmanslab/locpatterns). Briefly, this package uses the per-cell mean and standard deviation of the single-transcript localization features to first identify a number of different patterns, by clustering random subsets of cells, such that the clusters are most reproducible. In a second step, it determines the similarity of each single cell to each of the identified patterns.

Supervised machine learning can be further applied to classify cells with a distinctive subcellular localization of transcripts [23].

One convenient way to evaluate the basic computational quantification of the localization of transcript molecules is the median distance of all transcript molecules to the nucleus. Plotting the median of this single-cell readout for multiple genes should yield a bimodal distribution (Fig. 6A): transcripts, which become translated at the endoplasmic reticulum (ER), should have a shorter distance to the nucleus compared to transcripts with a cytoplasmic translation. For instance, we noticed that transcripts of RAB13, which have previously been described to enrich in filopodia [35], tended to be furthest from the nucleus (Fig. 6B). One way of controlling finer details of the localization of transcripts is the unbiased clustering of genes by multiple readouts

of the localization of transcripts. Mitochondrially-encoded transcripts should be identified as a group of colocalizing transcripts even when mitochondria are not stained [4]. Furthermore, at least in HeLa cells, one should observe a further sub-clustering of different groups of mitochondrially-encoded transcripts reflecting different positions within the mitochondria [4]. In addition, this may reveal further subclustering of transcripts translated at the ER [4], as well as transcripts translated in the cytoplasm. Such findings indicate extensive functional subcompartmentalization of the transcriptome, both on organelles and in the cytoplasm, which are properties of posttranscriptional control of gene expression that have remained hidden thus far.

4. Conclusion

Image-based transcriptomics combines precise counting of transcript molecules with a unique multivariate quantification of the subcellular position of each single transcript molecule for thousands of genes in tens of thousands of single cells. Being an image-based *in situ* technology it can be readily combined with image-based assays, which monitor additional specific biological markers of interest. To enable such lines of research, every experimental and computational step of image-based transcriptomics needs to be highly reproducible across different weeks and geared towards the quantification of single molecules. To enable image-based transcriptomics to reach its full potential, we developed computer vision algorithms that build on and improve those currently used to detect objects in confocal images. By using iterative watershedding we have improved the segmentations of nuclei and cells. In addition, we describe how to perform spot detection for transcript identification in an automated way for thousands of images. Accurate detection of nuclear outlines, cell outlines, and transcript molecules are essential for the correct quantification of a high-dimensional multivariate feature space of each transcript and to reveal bona fide novel properties of the spatial organization of the transcriptome [4]. The computer vision pipeline presented here complements our earlier work [4], and can be used independently of transcripts in other image-based approaches. It also forms a practical guide on how to extend image-based-assays to mapping small particles relative to spatial hallmarks of single cells. Indeed, the highly robust and automated protocol of the underlying computer vision pipeline has been instrumental for uncovering parameters of gene expression, which remain otherwise hidden.

Fig. 1.: Outline of image-based transcriptomics using bDNA sm-FISH.

Fig. 2.: A, B) Method for illumination correction of images. For each channel the mean intensity μ_i and the standard deviation σ_i are calculated for each pixel p_i in the field of view. Then an overall mean intensity $\overline{\mu}$ as well as the mean standard deviation $\overline{\sigma}$ of all pixels is derived from the "mean" and "std." matrices. Illumination correction is performed by per-pixel z-scoring (eq. 1), where z_i is the z-scored value for pixel p_i and In_i is the original intensity value for pixel p_i in a given image. The corrected intensity value C_i for pixel p_i in an image was then calculated as in eq. 2. C) Illumination correction examples for the DAPI channel. D) As in C but for Alexa Fluor succinimidyl ester (a general protein stain).

Fig. 3.: A) Scheme for nuclei segmentation and iterative correction of primary segmented objects. B) Strategy for selection of objects to be separated by combining the object solidity, form factor and area. All features measured as in the CellProfiler module "MeasureObjectAreaShape.m"

Fig. 4.: Improvement of segmentation of cells by iterative correction. Several different and partially overlapping segmentations are combined to a single optimal segmentation (Panel A). Detection of single cells stained by carboxylic acid, succinimidyl ester (Panel B).

Fig. 5.: Detection of single transcripts as spots. Application of a Laplacian of Gaussian (LoG) filter emphasizes spot-like signals (Panel A). Workflow for detecting transcripts as spots (Panel B). The specific numerical value for the detection threshold only mildly affects the number of spots once the intensities of individual images are rescaled similarly. Lines represent five different, randomly chosen images; arrows and asterisk indicate suggested thresholds (Panel C). The signal of individual transcripts is slightly teared in the corner of an image (Panel D).

Fig. 6.: Readouts of single-transcript localization (Panel A). Pipeline of converting single-transcript readouts to single-cell readouts (Panel B). Inspecting expected behavior of basic measurements of the localization of transcripts. The distance of transcripts to the nucleus is shorter for transcripts translated at the ER (Panel C). Median distance of all transcripts is normalized by z-scoring against 100 relocalizations of the transcripts to random pixels of the cytoplasm. Median of all cells over all single-cell medians is shown (Panel D). Differing distances to the nucleus become apparent to humans in large cells upon visualizing transcripts (green), the nucleus (blue) and the cell outline (white lines) (Panel D, numbers as in Panel C).

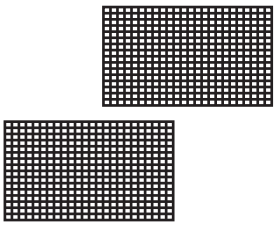
[1] Tang, F. *et al.* mRNA-Seq whole-transcriptome analysis of a single cell. Nature methods (2009) 6, 377-382.
[2] Islam, S. *et al.* Characterization of the single-cell transcriptional landscape by highly multiplex RNA-seq. Genome Res (2011) 21, 1160-1167.
[3] Hashimshony, T., Wagner, F., Sher, N. & Yanai, I. CEL-Seq: single-cell RNA-Seq by multiplexed linear amplification. Cell reports (2012) 2, 666-673.
[4] Battich, N., Stoeger, T. & Pelkmans, L. Image-based transcriptomics in thousands of single human cells at single-molecule resolution. Nature methods (2013) 10, 1127-1133.
[5] Picelli, S. *et al.* Full-length RNA-seq from single cells using Smart-seq2. Nat Protoc (2014) 9, 171-181.
[6] Wu, A. R. *et al.* Quantitative assessment of single-cell RNA-sequencing methods. Nature methods (2014) 11, 41-46.
[7] Fan, H. C., Fu, G. K. & Fodor, S. P. Expression profiling. Combinatorial labeling of single cells for gene expression cytometry. Science (2015) 347, 1258367.
[8] Liberali, P., Snijder, B. & Pelkmans, L. Single-cell and multivariate approaches in genetic perturbation screens. Nature reviews. Genetics (2015) 16, 18-32.
[9] Crosetto, N., Bienko, M. & van Oudenaarden, A. Spatially resolved transcriptomics and beyond. Nature reviews. Genetics (2015) 16, 57-66.
[10] Sinnamon, J. R. & Czaplinski, K. Locating RNAs in situ with FISH-STIC probes. Methods in molecular biology (2015) 1206, 137-148.
[11] Wang, F. *et al.* RNAscope: a novel in situ RNA analysis platform for formalin-fixed, paraffin-embedded tissues. The Journal of molecular diagnostics : JMD (2012) 14, 22-29.
[12] Cabili, M. N. *et al.* Localization and abundance analysis of human lncRNAs at single-cell and single-molecule resolution. Genome biology (2015).
[13] Ke, R. *et al.* In situ sequencing for RNA analysis in preserved tissue and cells. Nature methods (2013) 10, 857-860.
[14] Lee, J. H. *et al.* Highly multiplexed subcellular RNA sequencing in situ. Science (2014) 343, 1360-1363.
[15] Lee, J. H. *et al.* Fluorescent in situ sequencing (FISSEQ) of RNA for gene expression profiling in intact cells and tissues. Nat Protoc (2015) 10, 442-458.
[16] Larsson, C., Grundberg, I., Soderberg, O. & Nilsson, M. In situ detection and genotyping of individual mRNA molecules. Nature methods (2010) 7, 395-397.
[17] Raj, A., van den Bogaard, P., Rifkin, S. A., van Oudenaarden, A. & Tyagi, S. Imaging individual mRNA molecules using multiple singly labeled probes. Nature methods (2008) 5, 877-879.
[18] Shapiro, E., Biezuner, T. & Linnarsson, S. Single-cell sequencing-based technologies will revolutionize whole-organism science. Nature reviews. Genetics (2013) 14, 618-630.
[19] Shalek, A. K. *et al.* Single-cell RNA-seq reveals dynamic paracrine control of cellular variation. Nature (2014) 510, 363-369.
[20] Buchser, W. *et al.* in *Assay Guidance Manual* (eds G. S. Sittampalam *et al.*) (2004).
[21] Snijder, B. *et al.* Population context determines cell-to-cell variability in endocytosis and virus infection. Nature (2009) 461, 520-523.
[22] Lundholt, B. K., Scudder, K. M. & Pagliaro, L. A simple technique for reducing edge effect in cell-based assays. Journal of biomolecular screening (2003) 8, 566-570.
[23] Ramo, P., Sacher, R., Snijder, B., Begemann, B. & Pelkmans, L. CellClassifier: supervised learning of cellular phenotypes. Bioinformatics (2009) 25, 3028-3030.

[24] Jones, T. R. *et al.* CellProfiler Analyst: data exploration and analysis software for complex image-based screens. BMC bioinformatics (2008) 9, 482.
[25] Vincent, L. & Soille, P. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. IEEE Transactions on Pattern Analysis & Machine Intelligence (1991) 13, 383-598.
[26] Otsu, N. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics (1979) 9, 62-66.
[27] Dijkstra, E. W. A note on two problems in connexion with graphs. Numerische Mathematik (1959) 1, 269-271.
[28] Carpenter, A. E. *et al.* CellProfiler: image analysis software for identifying and quantifying cell phenotypes. Genome biology (2006) 7, R100.
[29] Rifkin, S. A. Identifying fluorescently labeled single molecules in image stacks using machine learning. Methods in molecular biology (2011) 772, 329-348.
[30] Trcek, T. *et al.* Single-mRNA counting using fluorescent in situ hybridization in budding yeast. Nat Protoc (2012) 7, 408-419.
[31] Mueller, F. *et al.* FISH-quant: automatic counting of transcripts in 3D FISH images. Nature methods (2013) 10, 277-278.
[32] Matas, J., Chum, O., Urban, M. & Pajdla, T. Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing (2002) 22, 761-767.
[33] Ruusuvuori, P. *et al.* Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images. BMC bioinformatics (2010) 11, 248.
[34] Bertin, E. & Arnouts, S. SExtractor: Software for source extraction. Astron Astrophys Sup (1996) 117, 393-404.
[35] Mili, S., Moissoglu, K. & Macara, I. G. Genome-wide screen reveals APC-associated RNAs enriched in cell protrusions. Nature (2008) 453, 115-119.

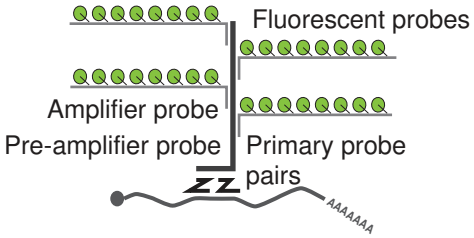
Acknowledgements

We would like to acknowledge A. Schwab for help on the development of the IdentifyPrimaryIterative.m module, Q. Nguyen and S. Lai from Affymetrix for helpful comments on experimental procedures, and V. Green for useful comments on the manuscript. L.P. acknowledges financial support for this project from the Swiss National Science Foundation, the University of Zurich and the University of Zurich Research Priority Program in Systems Biology and Functional Genomics.

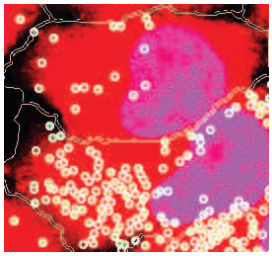
Figure 1



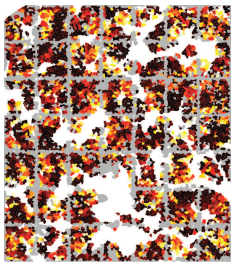
Cultivate cells and experiment in 384 well plates



Branched DNA single-molecule Fluorescence *in situ* hybridization



Detect single transcript molecules and organelles



Quantify transcripts in thousands of single cells

Figure 2

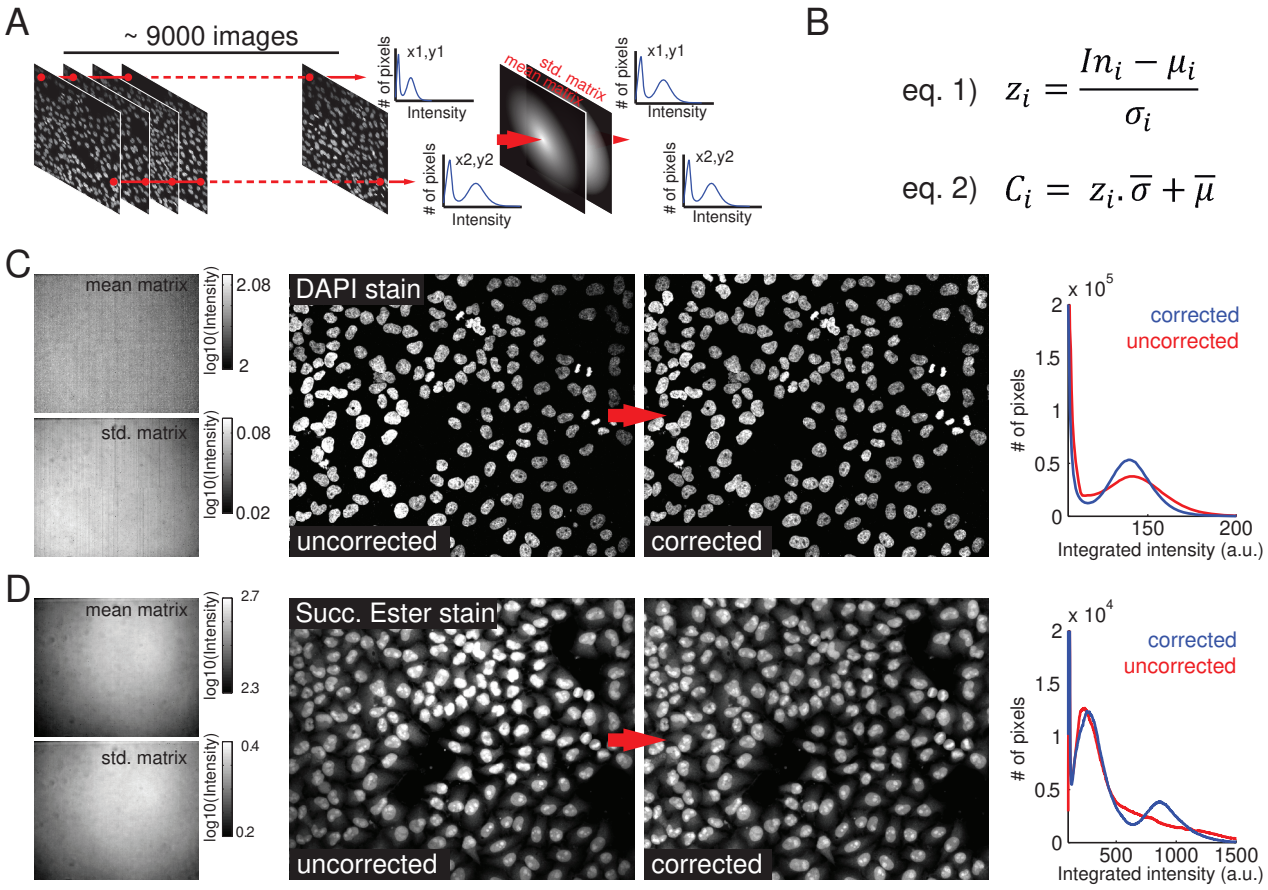
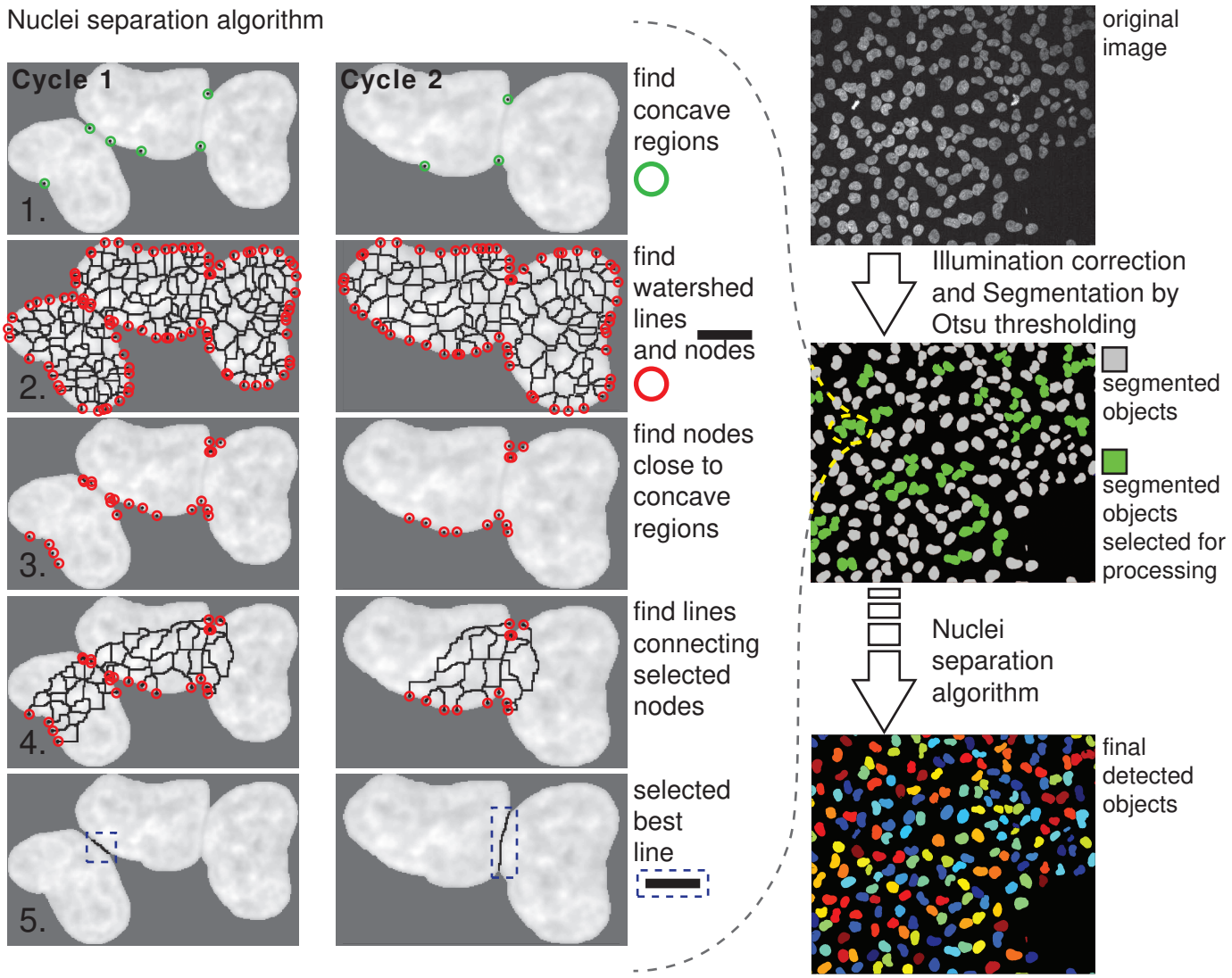

A
~ 9000 images
# of pixels
x1,y1
Intensity
std. matrix
mean matrix
x2,y2
B
eq. 1) $z_i = \frac{In_i - \mu_i}{\sigma_i}$
eq. 2) $C_i = z_i . \overline{\sigma} + \overline{\mu}$
C
mean matrix
std. matrix
log10(Intensity)
2.08
2
0.08
0.02
DAPI stain
uncorrected
corrected
x 10^5
corrected
uncorrected
# of pixels
Integrated intensity (a.u.)
D
mean matrix
std. matrix
2.7
2.3
0.4
0.2
Succ. Ester stain
uncorrected
corrected
x 10^4
corrected
uncorrected
# of pixels
Integrated intensity (a.u.)

Figure 3

A


Nuclei separation algorithm
Cycle 1
Cycle 2
1.
2.
3.
4.
5.
find concave regions
find watershed lines and nodes
find nodes close to concave regions
find lines connecting selected nodes
selected best line
original image
Illumination correction and Segmentation by Otsu thresholding
segmented objects
segmented objects selected for processing
Nuclei separation algorithm
final detected objects

B

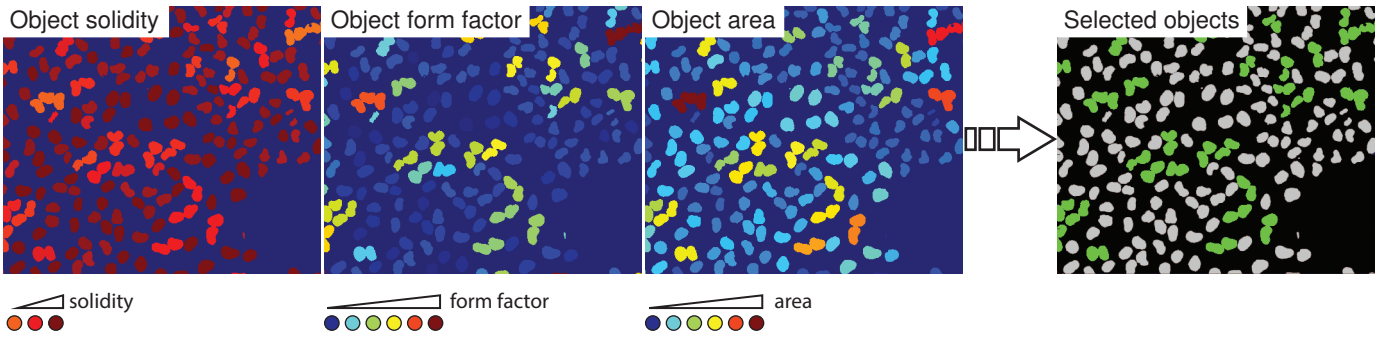
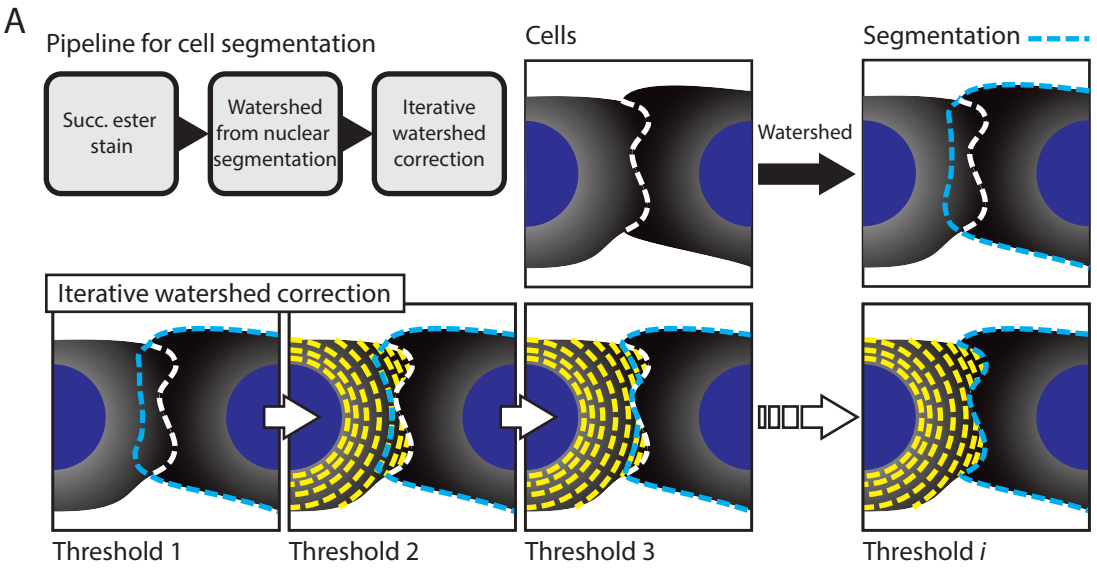

Object solidity
Object form factor
Object area
Selected objects
solidity
form factor
area

Figure 4


A
Pipeline for cell segmentation
Succ. ester stain
Watershed from nuclear segmentation
Iterative watershed correction
Cells
Segmentation
Watershed
Iterative watershed correction
Threshold 1
Threshold 2
Threshold 3
Threshold i

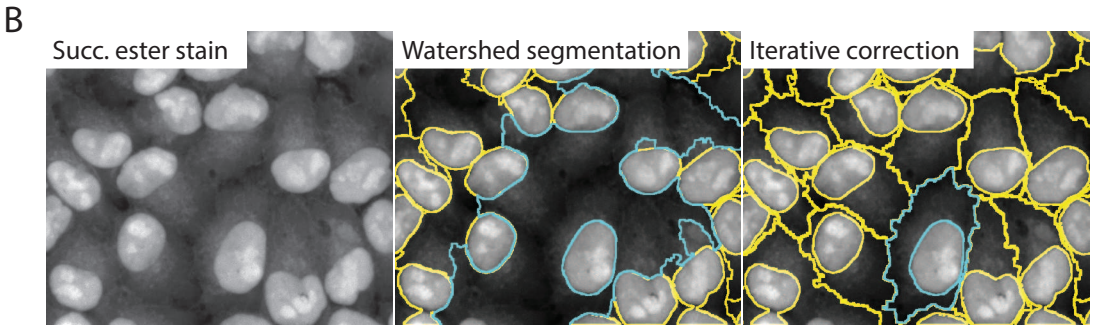
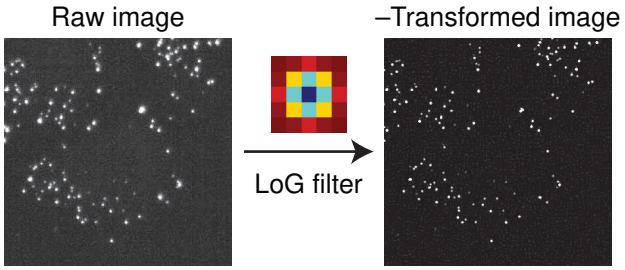

B
Succ. ester stain
Watershed segmentation
Iterative correction

Figure 5

A

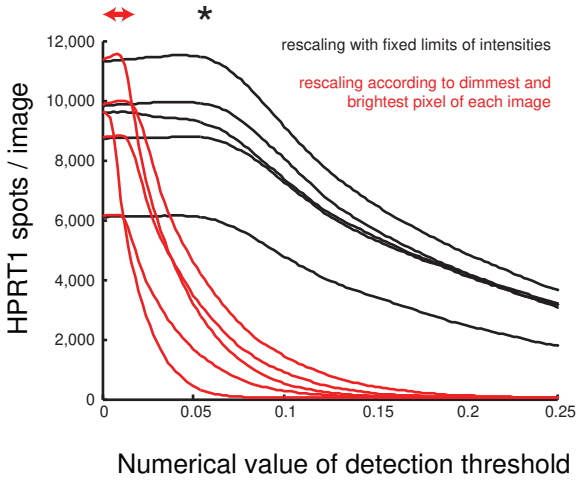


B

Step	Considered images	Computation
Obtain range of single-pixel intensities	4 reference wells of dapB and HPRT1	Local or cluster
Compute spots/image for different thresholds	4 reference wells of dapB and HPRT1	Local or cluster
Manually choose robust threshold	4 reference wells of dapB and HPRT1	Local
Count spots at the position of every pixel	all images of a plate	Cluster via CellProfiler
Compute correction matrix	all images of a plate	Local
Detect transcripts as spots	all images of a plate	Cluster via CellProfiler
Control results of detection	all images of a plate	Local

Essential Optional

C



D

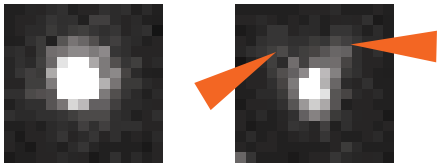
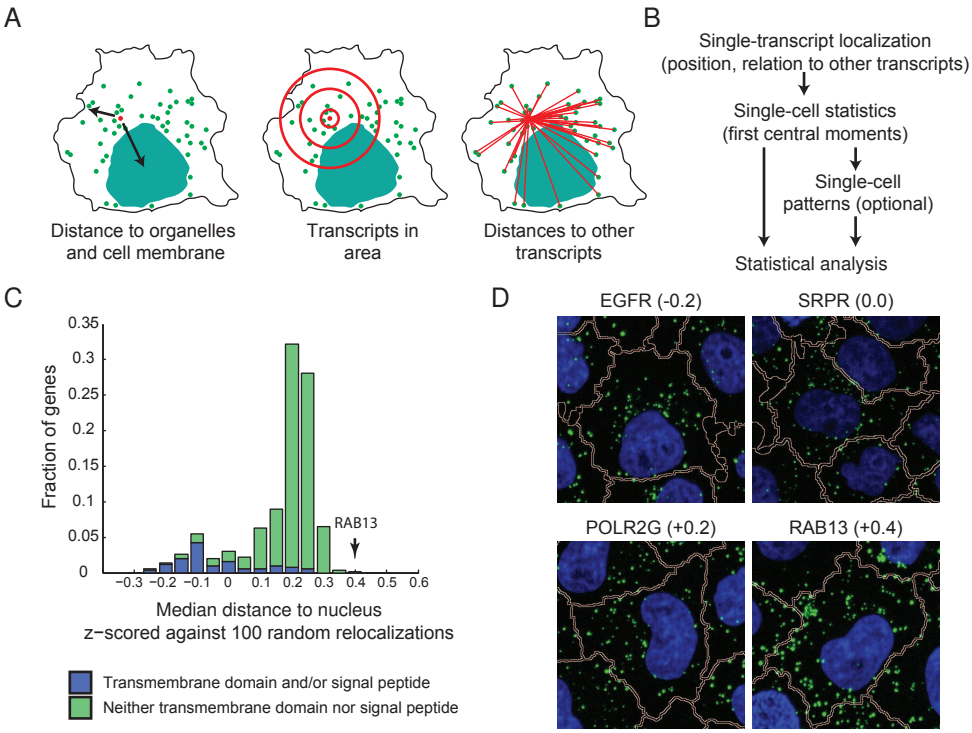


Figure 6


A
Distance to organelles and cell membrane
Transcripts in area
Distances to other transcripts
B
Single-transcript localization (position, relation to other transcripts)
Single-cell statistics (first central moments)
Single-cell patterns (optional)
Statistical analysis
C
Fraction of genes
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
RAB13
−0.3 −0.2 −0.1 0 0.1 0.2 0.3 0.4 0.5 0.6
Median distance to nucleus
z−scored against 100 random relocalizations
Transmembrane domain and/or signal peptide
Neither transmembrane domain nor signal peptide
D
EGFR (-0.2)
SRPR (0.0)
POLR2G (+0.2)
RAB13 (+0.4)